



University of Groningen

Fifth International Workshop on Sharing and Reusing Architectural Knowledge (SHARK 2010)

Lago, Patricia; Avgeriou, Paris; Kruchten, Philippe

Published in:
EPRINTS-BOOK-TITLE

IMPORTANT NOTE: You are advised to consult the publisher's version (publisher's PDF) if you wish to cite from it. Please check the document version below.

Document Version
Publisher's PDF, also known as Version of record

Publication date:
2010

[Link to publication in University of Groningen/UMCG research database](#)

Citation for published version (APA):

Lago, P., Avgeriou, P., & Kruchten, P. (2010). Fifth International Workshop on Sharing and Reusing Architectural Knowledge (SHARK 2010). In EPRINTS-BOOK-TITLE University of Groningen, Johann Bernoulli Institute for Mathematics and Computer Science.

Copyright

Other than for strictly personal use, it is not permitted to download or to forward/distribute the text or part of it without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license (like Creative Commons).

Take-down policy

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

Downloaded from the University of Groningen/UMCG research database (Pure): <http://www.rug.nl/research/portal>. For technical reasons the number of authors shown on this cover page is limited to 10 maximum.

Fifth International Workshop on Sharing and Reusing Architectural Knowledge (SHARK 2010)

Patricia Lago
VU University Amsterdam,
The Netherlands
patricia@cs.vu.nl

Paris Avgeriou
University of Groningen,
The Netherlands
paris@cs.rug.nl

Philippe Kruchten
University of British
Columbia, Canada
pbk@ece.ubc.ca

Abstract

Architectural Knowledge (AK) is defined as the integrated representation of the software architecture of a software-intensive system or family of systems along with architectural decisions and their rationale, external influence and the development environment. The SHARK workshop series focuses on current methods, languages, and tools that can be used to extract, represent, share, apply, and reuse AK, and the experimentation and/or exploitation thereof. This fifth edition of SHARK will discuss, among other topics, the contributions of this community to a Body of Knowledge on software architecture.

Categories and Subject Descriptors

D.2.11 [Software Engineering]: Software Architecture.

General Terms

Management, Documentation, Design, Human Factors, Standardization, Languages, Theory.

Keywords

Software architecture, Knowledge management.

1. Theme and goals

Software architecture is crucial to manage the complex interactions and dependencies between the stakeholders and to provide a central artifact that can be used for reference by them. Modern notational and

documentation approaches to software architecture shifted their focus from plain components and connectors to the design decisions that resulted in the architecture as well as the organizational, process and business rationale underlying them.

This workshop focuses on current architectural knowledge (AK) management approaches: methods, languages, notations, tools to extract, represent, and share AK.

The AK community is comprised of both researchers and industrial practitioners that are involved in a wide variety of fields, disciplines and application domains [1]. The SHARK workshop is a meeting place for this community, which has grown and matured over the last five years. In its past editions, the workshop explored the state of the art and practice in the field [2], a research agenda driving future R&D [3], and emerging approaches of AK in a broader context [4].

In this fifth edition of SHARK [5], we've proposed to the SHARK community to discuss and contribute on how to reorganize and codify a Body of Knowledge of the WICSA community (WICSA BOK). This BOK is partially available through Software Architecture Portal [6] and www.wicsa.net, but it needs to be reorganized and unified.

We see two broad objectives: (1) to codify the BOK in the way the potential users (the members of the architecture community) would like to see it; and (2) exploit Web 2.0 and social networking techniques to implement personalization, and better reachability/usability according to the needs of the community. SHARK contributors will have the opportunity to propose their ideas and R&D results to shape the next generation BOK from the Software Architecture Portal.

3. Accepted Papers

Out of 25 submissions we have accepted 13, covering a wider range of topics than we anticipated, falling roughly in 4 categories: Modeling and visualizing architectural decisions; Creation of AK,

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
ICSE'10, May 2–8, 2010, Cape Town, South Africa.
Copyright © 2010 ACM 978-1-60558-719-6/10/05 ... \$10.00.

using business goal, or rationale; Using AK for supporting the evolution of systems; Tools for sharing AK; all feeding nicely into various entries of the BOK.

Wang & Burge propose an approach to assist the architecture design process by selecting appropriate architecture patterns, to satisfy the system's NFRs. The approach is embedded in an AK management tool-suite, hence capturing architecture design decisions and their rationale with respect to the pattern selection.

Clements & Bass argue that business goals are an important type of AK, as they derive the quality attributes that drive architectural design. They introduce a format for describing business goals as scenarios, some standard categories of business goals and a method to help their elicitation and incorporation in the architecting process.

Nowak et al. state a number of research challenges about architecture decision modeling and with respect to the following aspects: a) knowledge capturing; b) knowledge sharing, c) evolution of architecture knowledge models, and d) support for the architectural design process. In each aspect they identify potential solutions and directions for future research.

Noppen & Tamzalit propose a framework for assisting architects to analyze AK during evolution activities. The framework allows the definition of desired system properties considered during evolution and the estimation of their relevance to the target architecture according to their accompanying AK.

Nakagawa et al. propose to use the Systematic Mapping technique from Evidence-Based Software Engineering to systematically identify, select, analyze and aggregate AK. They present a process for doing so and an example showing preliminary but promising results.

Clerc & de Vries discuss the extent to which AK management practices in global software development can be supported by a set of generic wiki functionalities distilled from the literature. This mapping provides guidance for the selection of wikis for AK management.

Mohamed & Zulkernine provide a taxonomy of architecture-based reliability efforts, classifying them according to reliability goals, component abstraction, and the level of granularity. The taxonomy has the potential to help the selection of architecture-based reliability techniques from existing ones, and serve as a stage for proposing novel or improved techniques.

Based on interviews with practitioners, *Ozkaya et al.* discuss the use of AK management in system evolution. The findings reveal that practitioners apply

architectural practices and do create and use AK; still, the lack of early quality attribute reasoning early on, and during system evolution, is a key contributor to failing to use AK effectively.

Brøndum & Zhu address the problem of capturing relations between systems in 'systems of software-intensive systems'(S3) and propose an 'S3 viewpoint' to capture knowledge about such relationships. In doing that, they provide an extensible taxonomy of system relationships, and analysis support.

Nikolaidou & Anagnostopoulos propose a number of model-based viewpoints for an enterprise information system, where the models are expressed in SysML. The viewpoint on evaluation is new, and it focuses on the simulation of the performance of the system. The various viewpoints are illustrated by a case study.

Shahin et al. present a survey on tools for architectural decision visualization and their use of the OSS tool Compendium to address the issues found in other tools.

Holmes et al. present a way to capture compliance-related AK for service-oriented architectures, in particular the rationale, following a model-driven approach to facilitate this process.

Abi-Antoun et al. report on how an experienced developer think in terms of objects and their relationships; they describe a field study where they evaluated the SCHOLIA approach, that extracts the run-time architecture of a software system.

References

- [1] Lago, P.; Avgeriou, P.; Capilla, R. & Kruchten, P. Wishes and Boundaries for a Software Architecture Knowledge Community: Proceedings of the Working IEEE/IFIP Conference on Software Architecture, IEEE Computer Society, 2008, 271-274
- [2] Lago, P. & Avgeriou, P. First workshop on sharing and reusing architectural knowledge SIGSOFT Softw. Eng. Notes, ACM Press, 2006, 31, 32-36.
- [3] Avgeriou, P.; Kruchten, P.; Lago, P.; Grisham, P. & Perry, D. Architectural knowledge and rationale: issues, trends, challenges, SIGSOFT Softw. Eng. Notes, ACM Press, 2007, 32, 41-46
- [4] Avgeriou, P.; Lago, P.; Kruchten, P. Towards Using Architectural Knowledge, SIGSOFT Softw. Eng. Notes, ACM Press, 2009, 34, 4 pp.
- [5] SHARK web site: www.cs.rug.nl/~paris/SHARK2010
- [6] The Software Architecture Portal at www.softwarearchitectureportal.org